Build Security In Home

What is Build Security In?

Build Security In (BSI) contains and links to best practices, tools, guidelines, rules, principles, and other resources that software developers, architects, and security practitioners can use to build security into software in every phase of its development. BSI content is based on the principle that software security is fundamentally a software engineering problem and must be addressed in a systematic way throughout the software development life cycle.

Build Security In is a project of the Software Assurance[1] program of the Strategic Initiatives Branch of the National Cyber Security Division (NCSD) of the U.S. Department of Homeland Security. The Software Engineering Institute (SEI) was engaged by the NCSD to provide support in the Process and Technology focus areas of this initiative. The SEI team and other contributors develop and collect software assurance and software security[2] information that helps to create secure systems.

How Can I Collaborate?

If you are new to the site, you will want to register to collaborate with other developers faced with the challenges of developing secure code. Register Now...

Call for Authors and Reviewers

Submit an article for publication on BSI or volunteer to review new articles. See the Call for Authors and Reviewers[4] for details.

What's New

Software Assurance (SwA) in Acquisition: Mitigating Risks to the Enterprise[5], prepared by members of the U.S. Department of Defense and Department of Homeland Security Software Assurance Acquisition Working Group, is available for download and comment. Comments must be submitted by November 20, 2007.

A DACS state-of-the-art report, Software Project Management for Software Assurance[6], is available for download and comment. Comments must be submitted by October 31, 2007.

A new state-of-the-art report published by the Information Assurance Technology Analysis Center, Software Security Assurance[7], is available for download.

Does your organization collect cost data that is specific to software assurance effort associated with software development? If so, what activities do you include and/or how do you reflect software

1. daisy:547 (Introduction to Software Security)
4. daisy:900 (Call for Authors and Reviewers)
5. daisy:908 (Software Assurance (SwA) in Acquisition: Mitigating Risks to the Enterprise)
6. daisy:906 (Software Project Management for Software Assurance: DACS State of the Art Report)
7. daisy:902 (SOAR on Software Security Assurance)

assurance activities in your work breakdown structure? Note that for our purposes we are not looking at activities that are normally accounted for in cost of operations, such as patch management, but at those associated with software assurance, such as threat modeling and analysis, development of misuse cases, risk analysis specifically for software assurance, etc. Please send replies to bsiauthor-request@cert.org[8].

BSI Updates

A newsletter describing additions to and significant revisions of BSI content is emailed periodically to subscribers. Follow the instructions[9] to subscribe or unsubscribe.

Process Agnostic Approach

BSI articles are grouped in a process agnostic view[10]. The content areas are classified in the following sections: Requirements, Architecture & Design, Code, Test, System, Management, and Fundamentals. Click on the thumbnail graphic at right to access navigation by process category.

11

Ten Most Recently Modified Articles

Name	Content Areas	Version Creation Time	Abstract

3. /daisy/registration?returnTo=%2Fdaisy%2Fbsi%2F2.html
8. mailto:bsiauthor-request@cert.org
9. daisy:704 (BSI Updates List)
10. daisy:438 (Process Agnostic Navigational View)
11. daisy:438 (Process Agnostic Navigational View)

Optimizing Investments in Security Countermeasures: A Practical Tool for Fixed Budgets	best-practices/requirements	9/11/07 11:19:39 AM	Software engineers and businesses must make the difficult decision of how much of their budget to spend on software security mitigation for the applications and networks on which they depend. This article introduces a novel method of optimizing using integer programming (IP), the combination of security countermeasures to implement to maximize system security under fixed resources. The steps in the method and recent results with a case study client are described.
Architectural Risk Analysis	best-practices/architecture	8/30/07 12:23:23 PM	Architectural risk assessment is a risk management process that identifies flaws in a software architecture and determines risks to business information assets that result from those flaws. Through the process of architectural risk assessment, flaws are found that expose information assets to risk, risks are prioritized based on their impact to the business, mitigations for those risks are developed and implemented, and the software is reassessed to determine the efficacy of the mitigations.
Arguing Security - Creating Security Assurance Cases	knowledge/assurance	7/31/07 12:31:13 PM	An assurance case is a body of evidence organized into an argument demonstrating that some claim about a system holds, i.e., is assured. An assurance

			case is needed when it is important to show that a system exhibits some complex property such as safety, security, or reliability. In this article, our objective is to explain an approach to documenting an assurance case for system security, i.e., a *security* assurance case or, more succinctly, a *security case*.
Acquisition Overview: The Challenges	best-practices/acquisition	6/5/07 11:08:59 AM	The challenges of acquiring software-intensive systems continue to grow along with the increasingly critical role software plays in supporting commercial and government enterprise, business, and mission needs. In addition to expanding functionality and complexity, mounting expectations for software systems to be flexible and interoperable add to acquisition challenges, notably in terms of ensuring their security.
Assuring Software Systems Security: Life Cycle Considerations for Government Acquisitions	best-practices/acquisition	6/5/07 10:53:28 AM	When systems are built under government contract, the acquirer and contractor share responsibility for the outcome, not only in terms of cost, schedule, and performance, but also with respect to quality attributes such as security. Using an acquisition life cycle framework, this article identifies acquirer activities, products, and resources that are necessary to establish

			and support contractor efforts to build secure software-intensive systems.
Building Security into the Business Acquisition Process	best-practices/acquisition	6/5/07 10:36:41 AM	This article presents the standard process for acquiring software products and services in business. It is based on the recommendations of the Agreement processes specified by the IEEE 12207 Standard. This standard presents the commonly accepted practices for ensuring a well-defined and persistent assurance process for acquired software. With the help of 12207, it is possible to integrate best practice in acquisition and supply into a single uniform approach. That approach will guarantee that security considerations will be a central part of product selection, monitoring, and acceptance. The ensuing set of policies and procedures provides rational control over all aspects of the process of securing acquired products. Properly followed, they will ensure an adequately secure software deliverable.
System Strategies References	best-practices/system-strategies	5/31/07 1:16:59 PM	System Strategies bibliography.
Plan, Do, Check, Act	best-practices/deployment	5/21/07 3:51:24 PM	This article describes a tried and true approach to security improvement that can be effectively used during deployment and operations. It identifies prerequisites

			that must be in place to sustain a desired state of security. It provides a set of minimum requirements for security hygiene and several security implementation frameworks that can be used in concert with the other articles in this content area.
Introduction to System Strategies	best-practices/system-strategies	5/21/07 3:23:30 PM	Trustworthiness can no longer be predicted by building software systems from discrete, isolated pieces that address static requirements within planned cost and schedule. Each new or updated component joins an existing operational environment and must merge with that legacy to form an operational whole. Today's technology must support an operating environment that is driven by business goals and organizational needs instead of a predefined infrastructure that functions within established technology constraints. The operating environment can be geographically and managerially distributed and dynamically changing. Few businesses can stop to make changes and then restart. This introduction discusses the effects of the changing operational environment on the development of secure systems.

Scale: System Development Challenges	best-practices/system-strategies	5/21/07 3:22:33 PM	The usage and characteristics of large systems or systems of systems can challenge many current development assumptions. Vulnerability analysis has typically concentrated on vulnerabilities induced by errors in coding or in the interfaces among components. System interactions can also be a seedbed for vulnerabilities, however. This article describes software assurance challenges inherent in networked systems development and proposes a structured approach to analyzing potential system stresses using scenarios.